What Is Claimed Is:

1. A method for generating an application, comprising the steps of:

receiving a functional description of an application; and

automatically partitioning the functional description of the application into a plurality of modules based on parameterized criteria.

2. The method of claim 1, wherein the functional description of the application comprises a flowchart description.

3. The method of claim 1, wherein the functional description of the application comprises a markup description.

4. The method of claim 1, wherein the parameterized criteria comprises a measure of application latency.

5. The method of claim 4, wherein automatically partitioning comprises:

partitioning the functional description of the application into a plurality of different partitions; and

using the parameterized criteria to determine which partition, among the plurality of different partitions, provides a minimal application latency.

6. The method of claim 5, wherein the parameterized criteria comprises a cost function, which is based on transmission and compilation time for different size modules, as the measure of application latency.

7. The method of claim 5, wherein the parameterized criteria comprises a probability measure for determining a probability of a given path in a partition being traversed.

8. The method of claim 1, further comprising automatically generating application code for each module.

9. The method of claim 1, further comprising automatically generating a controller that can navigate between the modules of the application.

10. The method of claim 1, wherein the step of receiving comprises automatically fetching the functional description of the application from a persistent storage location; and

performing on-line dynamic remodularization of the application.

11. The method of claim 10, wherein performing on line dynamic remodularization comprises:

adapting the parameterized criteria according to changes in an environment in which the application is deployed; and

automatically partitioning the functional description of the application into a plurality of modules based on the adapted parameterized criteria.

12. A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for generating an application, the method steps comprising:

receiving a functional description of an application; and

automatically partitioning the functional description of the application into a plurality of modules based on parameterized criteria.

13. The program storage device of claim 12, wherein the functional description of the application comprises a flowchart description.

14. The program storage device of claim 12, wherein the functional description of the application comprises a markup description.

15. The program storage device of claim 12, wherein the parameterized criteria comprises a measure of application latency.

16. The program storage device of claim 15, wherein the instructions for automatically partitioning comprise instructions for:

partitioning the functional description of the application into a plurality of different partitions; and

using the parameterized criteria to determine which partition, among the plurality of different partitions, provides a minimal application latency.

17. The program storage device of claim 16, wherein the parameterized criteria comprises a cost function, which is based on transmission and compilation time for different size modules, as the measure of application latency.

18. The program storage device of claim 12, further comprising instructions for automatically generating application code for each module.

19. The program storage device of claim 12, further comprising instructions for automatically generating a controller that can navigate between the modules of the application.

20. The program storage device of claim 12, wherein the instructions for receiving comprise instructions for automatically fetching the functional description of the application from a persistent storage location; and

performing on-line dynamic remodularization of the application.

21. The program storage device of claim 20, wherein the instructions for performing on-line dynamic remodularization of the application comprise instructions for:

adapting the parameterized criteria according to changes in an environment in which the application is deployed; and

automatically partitioning the functional description of the application into a plurality of modules based on the adapted parameterized criteria.

22. A tool for generating an application, comprising:

an application partition module that automatically partitions a functional description of an application into a plurality of modules based on parameterized criteria.

23. The tool of claim 22, further comprising a user interface for enabling a user to generate a functional description of an application.

24. The tool of claim 22, wherein the functional description comprises a flowchart description.

25. The tool of claim 22, wherein the functional description comprises a markup description.

26. The tool of claim 22, wherein the parameterized criteria comprises a cost function for resource utilization in a computing environment in which the application is employed.

27. The tool of claim 26, wherein the computing environment comprises a network environment.

28. The tool of claim 22, wherein the parameterized criteria comprises a measure of application latency.

29. The tool of claim 28, wherein the parameterized criteria comprises a cost function, which is based on transmission and compilation time for different size application modules, as the measure of application latency in a network environment.

30. The tool of claim 22, further comprising a code generator that automatically generates application code for each module.

31. The tool of claim 30, wherein the code generator automatically generates a controller that can navigate between the modules of the application.

32. A method for generating an application, comprising the steps of:

receiving a functional description of a network application; and

automatically partitioning the functional description of the network application into a plurality of modules based on parameterized criteria of network latency.

33. The method of claim 32, wherein the functional description of the network application comprises a flowchart description or a markup description.

34. The method of claim 32, wherein the network application comprises a client-server application.

35. The method of claim 32, wherein the network application comprises a Web application.

36. The method of claim 32, wherein automatically partitioning comprises:

partitioning the functional description of the network application into a plurality of different partitions; and

using the parameterized criteria to determine which partition, among the plurality of different partitions, provides a minimal network latency.

37. The method of claim 32, wherein the parameterized criteria comprises a cost function, which is based on transmission and compilation time for different size modules, as the measure of network latency.

38. The method of claim 32, further comprising automatically generating application code.

39. The method of claim 38, wherein each module comprises an application view page, and wherein automatically generating application code comprises generating markup for each view page and generating server-side code for dispatching the view pages.

40. The method of claim 39, further comprising automatically generating a controller that can navigate between the view pages of the application.